Spatio-temporal Object Recognition

Roeland De Geest[1,3,4], Francis Deboeverie[2,3], Wilfried Philips[2,3], and Tinne Tuytelaars[1,3⋆]

[1] KU Leuven ESAT - PSI, Leuven, Belgium
[2] UGent TELIN - IPI, Ghent, Belgium
[3] iMinds
[4] Roeland.DeGeest@esat.kuleuven.be

Abstract. Object recognition in video is in most cases solved by extracting keyframes from the video and then applying still image recognition methods on these keyframes only. This procedure largely ignores the temporal dimension. Nevertheless, the way an object moves may hold valuable information on its class. Therefore, in this work, we analyze the effectiveness of different motion descriptors, originally developed for action recognition, in the context of action-invariant object recognition. We conclude that a higher classification accuracy can be obtained when motion descriptors (specifically, HOG and MBH around trajectories) are used in combination with standard static descriptors extracted from keyframes. Since currently no suitable dataset for this problem exists, we introduce two new datasets and make them publicly available.

1 Introduction

Object recognition is one of the main topics of interest in computer vision. In still images, it has been extensively covered (e.g., [4, 11]) and various competitions such as Pascal VOC [3] and ImageNet [16] have encouraged research and provided datasets for evaluation and comparison of the developed methods. Far fewer object recognition methods have been published for video data, however. In most cases (e.g., [17, 18]) keyframes are extracted from the video. These frames are chosen in such a way that they represent the whole video, preserving the information (i.e., the object appearance) in the video as much as possible. Afterwards, standard image object recognition methods are applied on these keyframes only. As a consequence, the video object recognition problem is reduced to static object recognition in the keyframes, and temporal information is left unexploited.

In this work, we investigate whether the motion of dynamic objects (and by extension, animals) can be used to improve their recognition. To this end, we evaluate object recognition methods that build on spatio-temporal representations, as typically used for action recognition. We start from two observations.

⋆ This work was financially supported by the project "Multi-camera human behavior monitoring and unusual event detection" (FWO G.0.398.11.N.10) and the PARIS project (IWT-SBO Nr. 110067).

First, video is a different domain than still images: Kalogeiton et al. [6] show that an object detector for video is best trained on video data as well. If we train on video anyway, it makes sense to exploit as much motion information as possible. Second, some object classes are dynamic: they change over time due to non-rigid deformations (e.g., a tree in the wind), manipulation (e.g., a driving car) or actions (e.g., a sleeping or walking lion). Because of this extra variability, these kinds of objects are more difficult to recognize. By using spatio-temporal features, we can capture these variations for a better recognition.

One issue that may have hampered the development of spatio-temporal object recognition methods is the lack of good datasets. Therefore, we introduce two new datasets for our experiments.

We start our discussion in the next section with a review of related work. In Section 3, we introduce our new datasets. Section 4 describes our experiments and the results are discussed in Section 5. Section 6 concludes the paper.

2 Related Work

Video object recognition There is only a limited amount of work on learning object detectors directly from video and/or applying them to videos as such (i.e., without falling back to the sampling of keyframes from the video and applying static detectors to these). While at first sight video seems a far richer format than still images, video comes with its own challenges, such as (typically) lower resolutions, interlacing effects, motion blur and compression artifacts. For these reasons, it has been argued that video is actually a different domain than still images [6, 15]. Instead of just applying classifiers trained on static images to video data, one should then compensate for the domain shift using domain adaptation methods.

Obviously, directly training models from video data avoids this issue. At the same time, as we argued earlier, this allows to exploit the richer information contained in video, including typical motion patterns and temporal continuity. Collecting ground truth annotations for video data is cumbersome, however, even when using specialized annotation tools such as [23]. Therefore, not many good datasets for learning object classifiers from video are available to date, except for action recognition (where the motion component is judged critical), surveillance (often with a static camera and hence limited background variation) or traffic sequences (mostly with very specific scene constraints that provide strong cues for detection). None of these seem suitable for evaluating object recognition methods exploiting motion cues in an unconstrained setting, as is the goal in this paper.

As one of a few exceptions that do train models directly from video data, we should mention the work of Viola et al. [19], who designed a pedestrian detector that uses both appearance and motion features. They calculate the absolute difference of the intensities of two consecutive frames and use it for their rectangle filters. This work is most closely related to our work. It starts with the same observation that motion is characteristic for the moving actor

or object. However, we use information over more frames and different, more modern descriptors. Moreover, we show the validity of this assumption for other, non-human, animals and objects.

Liu et al. [10] developed a method to recognize static, non-moving objects in videos. The representation of an object changes when the camera moves. They map these representations on a manifold and perform manifold-to-manifold matching for recognition. By using a moving camera, they get multiple views from the object and some insight in its depth structure. Their objects are static. We, however, focus on what can be learned from the motion of the object itself. Moreover, they only want to recognize specific objects, not object classes as we do. Finally, they test on videos with a homogenous background. We use realistic user-generated YouTube content.

A related problem is **gait recognition** (see [8] for a survey). In this topic, the aim is to recognize human individuals based on the way they walk. We, on the other hand, focus on category-level recognition.

Several methods have recently been proposed focusing on **video segmentation**, e.g., [1, 5]. These methods exploit motion information in video, either in the form of optical flow or through a set of points tracked over a small fragment of the video (trajectories), to discover which pixels belong to the same object. Most recently, Oneata et al. [14] proposed a 'tube' (bounding box of an object over time) detection method starting from a graph of superpixels. However, instead of recognizing particular object categories, these methods at best detect and locate unknown objects (and with low precision).

Prest et al. [15] build on the approach of [1] to find objects in videos and use these as additional data for training an object detector for images. We show results on the dataset they compiled from YouTube videos, but the reader should note its small scale, with only about ten example videos for some of the categories.

Action recognition has been intensively studied over the last years, and several powerful motion descriptors have been proposed in this context. In practice, the distinction between recognizing actions and recognizing objects can sometimes be diffuse: a *pedestrian* is often considered a different object category than the more general *person*. Likewise, one could consider *human playing tennis* and *human walking* as two different objects. However, this is a very constrained formulation. Objects and animals not only show variation between instances, but they can also perform multiple actions and therefore have multiple motion patterns. Learning a separate classifier for each object/action combination is not desirable as it further increases the amount of training data needed.

There are two main approaches for feature extraction in videos. In the first, the video is considered as a 3D image, with time as a third dimension, and 2D interest point detectors are extended to 3D (e.g., [2, 7, 22]). In the second approach, points are tracked over time and the resulting trajectories are used as features (e.g., [12, 13, 20]). In our experiments, we use the dense trajectories of Wang et al. [20].

3 Savanna Datasets

Since currently no suitable video object recognition dataset is publicly available, we collected our own data and intend to share it with anyone interested. We sampled 86 videos from YouTube with African animals, such as elephants, giraffes and lions. We divided these videos in three-second fragments (75 frames at a frame rate of 25 Hz). We select the fragments where only one species of animal is present (there can be multiple individuals of the same species, however), the animal or a part of it is moving and the camera focal length is not changed drastically (i.e., not heavily zooming in or out) to make tracking of (parts of) the animals easier.

The fragments are labeled according to their species. More videos are available for some animals than for others, but evaluation on a balanced dataset is easier to interpret. Therefore, we create two datasets. The first consists of approximately 100 fragments from each of four animal classes; we call it **Savanna4**. The second, **Savanna7**, has about 60 fragments from seven species. The species, number of videos and number of fragments for both datasets can be found in Table 1. For evaluation, we divide the fragments of a dataset in five groups with a similar number of fragments per class and perform leave-one-group-out cross validation. Fragments from the same YouTube video are kept in the same group to avoid contamination between fragments for training and fragments for testing. The final classification score is obtained by averaging the accuracies of all classes.

	Savanna4		Savanna7	
	Videos	Fragm.	Videos	Fragm.
Giraffe	21	106	21	60
Lion	12	99	12	60
Rhinoceros	15	90	15	60
Elephant	16	106	14	60
Antelope			14	61
Baboon			13	55
Zebra			9	47
Total	55	401	82	403

Table 1. Classes of the Savanna4 and Savanna7 datasets with their numbers of videos and fragments.

Recognition of the species in these datasets is challenging for multiple reasons. First, the animal is not always completely visible due to occlusion or bad image composition; on the other hand, some fragments contain complete herds. Second, the pose and activity of the animal can differ: some animals are standing or walking, others eating or drinking. Third, the appearance of animals varies within the species: a male lion is clearly different from the female or juvenile. Finally,

compression artefacts are clearly visible. The image quality is often significantly lower than what one finds in static image datasets. An extra challenge for video-based methods lies in the moving camera: most videos were recorded by amateurs on safari. While we avoid strong zooming effects, the camera may follow the animal, or may be unstable since it is hand held. Figure 1 shows some snapshots of fragments in our datasets.

Fig. 1. Frames from our Savanna datasets. The first four species are in both datasets, the last three only in Savanna7.

4 Experimental Setup

4.1 Datasets

We conduct experiments on four datasets. **Savanna4** and **Savanna7** are already described in Section 3. Our third dataset is based on the **Wild8** dataset from Liu et al. [9]. This dataset consists of 100 videos of African landscape and animals. It was collected for video object segmentation and has eight categories: bird, lion, elephant, sky, tree, grass, sand and water. Each video comprises three seconds at a sample rate of 10 Hz. Only the three animals move sufficiently to be considered for our application. This way, we have 47 videos of birds, 15 videos of lions and 11 videos of elephants. We split them in five groups to be able to evaluate with leave-one-group-out cross validation and still have a decent (albeit small) number of training examples. The final classification score is obtained by averaging the accuracies of all classes. Figure 2 shows some frames of the dataset.

Our fourth dataset is an adaptation of the **Youtube Objects** dataset collected by Prest et al. [15]. This dataset consists of videos (split in shots) of ten object and animal categories (examples in Fig. 3). We only know the object is present somewhere in each video. Therefore, we check for each shot whether the object is visible while no other classes of the dataset are; otherwise, the shot is removed. Next, we split the shots in 30-frame fragments. We divide the fragments in four groups. Fragments coming from the same video are in the same group.

Fig. 2. Example frames of the classes in the Wild8 dataset.

Here too, we evaluate with leave-one-group-out cross validation and average accuracy. Table 2 shows the classes with their numbers of videos and fragments. We should emphasize that some videos generate hundreds of fragments, while others have only one valid fragment. Videos with many fragments have a high influence on the trained model, even though it is plausible that they contain very limited motion and appearance information. In that case, the model does not generalize well and unseen object instances are classified wrongly.

Fig. 3. Example frames of the classes in the YouTube Objects dataset.

	Videos	Fragments		Videos	Fragments
Airplane	13	1854	Bird	16	784
Boat	17	2114	Cat	21	1303
Car	9	374	Cow	22	1679
Motorbike	14	1081	Dog	36	2696
Train	30	5873	Horse	29	4300

Table 2. Classes of the adapted YouTube Objects dataset with their numbers of videos and fragments.

4.2 Features and Classifier

The motion features we use are the HOG, HOF and MBH descriptors around a trajectory as well as the motion of the trajectory itself, as in [20]. We use their setup with one exception: to reduce the calculation time, we set the sampling step size to $W = 10$. Only on the Wild8 dataset, where the amount of data is limited, we take $W = 5$. We also experimented with improved trajectories [21] that try to compensate for the camera motion of the video. Our object recognition results were slightly better, but with a similar improvement (about 2%) in all possible settings. We kept using the basic dense trajectories for the rest of our experiments, because their effect has been more extensively researched in all kinds of video processing applications.

As a baseline, we implement a keyframe-based approach. For this, we calculate dense 128-bin SIFT descriptors [11] with the same spatial density W as the trajectories over the same eight scales. We take a frame every $L = 15$ frames. This way, we ensure that the number of SIFT descriptors is roughly equal to the maximum number of trajectories. In practice, however, on average many more SIFT descriptors than trajectories are extracted, since trajectories are not started in homogeneous regions and can still be discarded after they finish. Note that a SIFT detector instead of dense SIFT would not improve recognition results, since the background of the videos has enough texture to let the detector fire.

Next, we train a codebook for each descriptor type and collect the quantized descriptors in a bag-of-words representation. Finally, we train a multi-channel one-against-all support vector machine with χ^2-kernel as in [20]. The recognized object category is the one with the highest probability. We take the average of the accuracies of all classes as final performance criterion.

This is a very basic setup: higher accuracies can easily be obtained with more sophisticated methods. The advantage of this scheme is that features and descriptors have a high influence on the classification accuracy, therefore making it well-suited to examine the effectiveness of descriptors.

5 Results and Discussion

Table 3 shows the average classification accuracy for all datasets for multiple descriptor combinations. On Savanna4, Savanna7 and Wild8, we conduct the experiments five times and report the average performance and the standard deviation. We experiment only once on YouTube Objects, because calculations on this dataset are more time-consuming.

When we use only one descriptor, SIFT is a good choice. It has top performance on the Savanna4 and Wild8 datasets and decent scores on the others; moreover, it is fast to calculate. Of the motion-containing descriptors, HOG and MBH are the best. These two descriptors preserve some appearance information: HOG directly, MBH by focusing on the boundaries of a moving object. The trajectory descriptor, the uncoded motion of the trajectory, is the least effective descriptor, probably because it is most affected by the camera ego-motion.

	Savanna4	Savanna7	Wild8	YouTube Obj.
SIFT	61.9% ± 0.6%	41.6% ± 1.0%	**62.3%** ± 4.5%	44.3%
Trajectory	33.1% ± 2.0%	16.9% ± 1.2%	41.3% ± 1.6%	23.9%
HOG	61.9% ± 1.3%	48.4% ± 1.1%	53.5% ± 1.7%	36.0%
HOF	39.6% ± 1.3%	19.0% ± 0.8%	39.7% ± 2.6%	27.9%
MBH	52.2% ± 0.8%	36.1% ± 0.7%	47.8% ± 1.7%	45.3%
HOG+MBH	65.7% ± 1.3%	47.2% ± 2.7%	49.2% ± 4.8%	47.7%
All trajectory	60.1% ± 2.3%	40.8% ± 1.1%	50.4% ± 3.3%	51.4%
SIFT+HOG	65.2% ± 0.9%	44.9% ± 0.6%	59.0% ± 6.7%	52.8%
SIFT+MBH	67.0% ± 0.5%	46.0% ± 0.7%	54.9% ± 4.9%	55.9%
SIFT+HOG+MBH	**68.6%** ± 0.9%	**49.6%** ± 0.4%	56.7% ± 3.4%	**58.1%**
SIFT+All trajectory	66.6%, ± 0.7%	43.3%, ± 0.9%	49.8%, ± 2.0%	57.1%

Table 3. Average accuracy on Savanna4, Savanna7, Wild8 and YouTube Objects datasets for different descriptor combinations. 'All trajectory' is short for 'Trajectory+HOG+HOF+MBH'.

The classification accuracy increases with well-chosen descriptor combinations. Recognition with SIFT and either HOG or MBH is already better than SIFT alone. The best results are obtained by combining HOG, MBH and SIFT, with an improvement of 7% over the keyframe approach on the Savanna datasets and 14% on YouTube Objects. Configurations with the pure trajectory descriptors and HOF, however, yield lower results than the ones without them. On Wild8, the single SIFT descriptor works best. This dataset is too small to draw conclusions: the instability of the results is indicated by the large values of the standard deviation in Table 3.

Some objects are not suited for recognition by motion. Figure 4 shows the confusion matrix of the Savanna7 dataset with only SIFT and only MBH. Zebras are often confused with antelopes with MBH, but not with SIFT: the appearance is very discriminative here. On the other hand, antelopes have a score three times higher with MBH. The dataset includes seven species of antelope, making the appearance more variable, while the motion is still similar. The different types of cats, dogs and birds in YouTube Objects give rise to a similar effect.

To find out whether appearance or motion is more easily learned, we train models on the Savanna4 dataset with a varying number of training samples. The classification accuracies can be found in Fig. 5. All descriptors start levelling out around the same time, so we conclude there is no significant difference in learning difficulty.

Another interesting observation is that HOG and SIFT combined are better than either of them separately, though they are both based on appearance. The difference lies in two aspects. First, the HOG descriptor is only centered on the moving points, not on a static background. Second, the descriptor is constructed differently. A HOG descriptor has some time information, not only because it is calculated over the 15 frames of a trajectory, but also because it is subdivided in spatio-temporal cells. These cells manage to preserve more structure in the

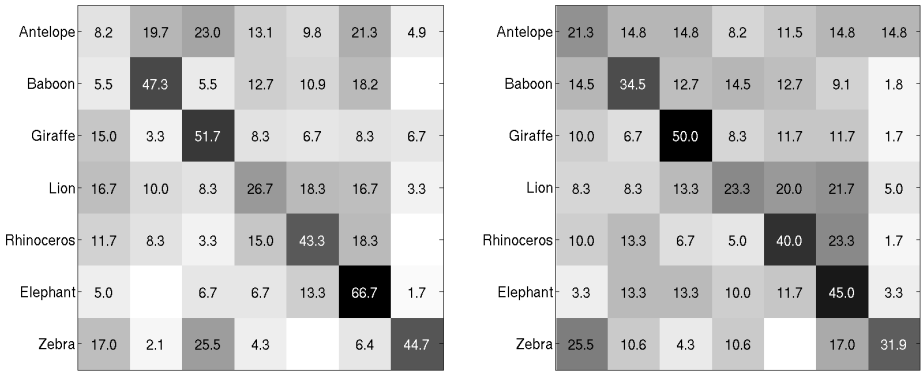

Fig. 4. Confusion matrices for Savanna7 dataset with SIFT (left) and MBH (right) descriptor.

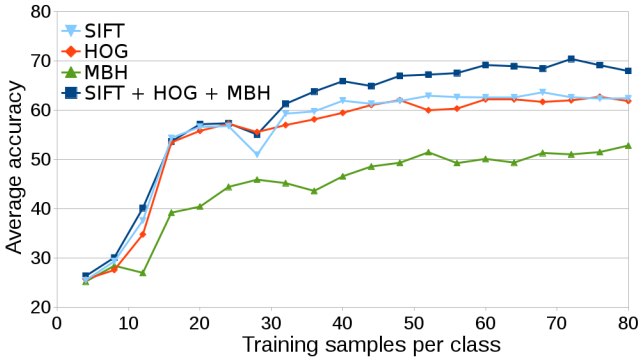

Fig. 5. Average classification accuracy as a function of the number of training samples per class for the Savanna4 dataset.

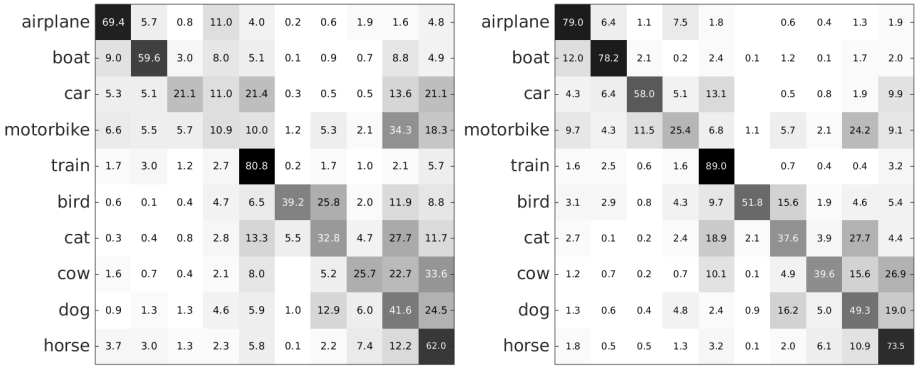

Fig. 6. Confusion matrix for the YouTube Objects dataset with SIFT (left) and combined descriptors HOG, MBH and SIFT (right).

HOG. The SIFT descriptor, however, has 32 more dimensions, so it can store its information at a finer scale. To investigate the effect of these two differences, we calculate a SIFT descriptor around the middle point of each trajectory and train a model on Savanna4 with these descriptors. The first difference is now neutralized. With this configuration, we obtain a score of 43.0%. This is significantly lower than the 61.9% of HOG and the 61.9% of standard SIFT. We can conclude that the main advantage of HOG is its structure-preserving descriptor, while SIFT makes efficiently use of a larger number of features.

Savanna4, Savanna7 and Wild8 contain only animal classes. YouTube Objects has some objects (means of transportation) as well, and for these too adding motion information benefits the recognition accuracy (as can be seen in the confusion matrices of Fig. 6). With SIFT descriptors, the average accuracy of the objects only is equal to 48.4%. When we add HOG and MBH, it increases to 65.9%. We observe four quadrants in the confusion matrix of the comined HOG, MBH and SIFT descriptors. Animals and means of transportation are less easily confused with each other than the different types of animal (or transportation) are with each other.

As a disadvantage, use of motion descriptors increases calculation time, mainly because the optical flow has to be calculated in order to track points and obtain MBH and HOF.

6 Conclusion

Motion is often discarded in video object recognition. We have shown that a higher accuracy can be obtained when it is taken into account. In particular, combining HOG and MBH descriptors around a trajectory with a standard dense SIFT method results in significantly higher performance than using only SIFT. We have introduced two new datasets and adapted two existing datasets to the

problem. These datasets will be made publicly available to stimulate and help further research on this topic.

References

1. Brox, T., Malik, J.: Object segmentation by long term analysis of point trajectories. In: Computer Vision–ECCV 2010, pp. 282–295. Springer (2010)
2. Dollár, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on. pp. 65–72. IEEE (2005)
3. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal visual object classes (VOC) challenge. International Journal of Computer Vision 88(2), 303–338 (2010)
4. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. pp. 1–8. IEEE (2008)
5. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph-based video segmentation. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. pp. 2141–2148. IEEE (2010)
6. Kalogeiton, V., Ferrari, V., Schmid, C.: Analysing domain shift factors between videos and images for object detection (2015)
7. Laptev, I., Lindeberg, T.: Space-time interest points. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on. pp. 432–439. IEEE (2003)
8. Liu, L.F., Jia, W., Zhu, Y.H.: Survey of gait recognition. In: Emerging Intelligent Computing Technology and Applications. With Aspects of Artificial Intelligence, Lecture Notes in Computer Science, vol. 5755, pp. 652–659. Springer (2009)
9. Liu, X., Tao, D., Song, M., Ruan, Y., Chen, C., Bu, J.: Weakly supervised multiclass video segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 57–64 (2013)
10. Liu, Y., Jang, Y., Woo, W., Kim, T.K.: Video-based object recognition using novel set-of-sets representations. In: Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on. pp. 533–540 (2014)
11. Lowe, D.G.: Object recognition from local scale-invariant features. In: Computer vision, 1999. The proceedings of the seventh IEEE international conference on. vol. 2, pp. 1150–1157. IEEE (1999)
12. Matikainen, P., Hebert, M., Sukthankar, R.: Trajectons: Action recognition through the motion analysis of tracked features. In: Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on. pp. 514–521. IEEE (2009)
13. Messing, R., Pal, C., Kautz, H.: Activity recognition using the velocity histories of tracked keypoints. In: Computer Vision, 2009 IEEE 12th International Conference on. pp. 104–111. IEEE (2009)
14. Oneata, D., Revaud, J., Verbeek, J., Schmid, C.: Spatio-temporal object detection proposals. In: Computer Vision–ECCV 2014, pp. 737–752. Springer (2014)
15. Prest, A., Leistner, C., Civera, J., Schmid, C., Ferrari, V.: Learning object class detectors from weakly annotated video. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3282–3289 (June 2012)

16. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge (2014)
17. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on. pp. 1470–1477. IEEE (2003)
18. Snoek, C., Sande, K., Rooij, O., Huurnink, B., Uijlings, J., Liempt, M., Bugalhoy, M., Trancosoy, I., Yan, F., Tahir, M., et al.: The MediaMill TRECVID 2009 semantic video search engine. In: TRECVID workshop (2009)
19. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on. pp. 734–741. IEEE (2003)
20. Wang, H., Klaser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 3169–3176. IEEE (2011)
21. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: Computer Vision (ICCV), 2013 IEEE International Conference on. pp. 3551–3558. IEEE (2013)
22. Willems, G., Tuytelaars, T., Van Gool, L.: An efficient dense and scale-invariant spatio-temporal interest point detector. In: Computer Vision–ECCV 2008, pp. 650–663. Springer (2008)
23. Yuen, J., Russell, B., Liu, C., Torralba, A.: Labelme video: Building a video database with human annotations. In: Computer Vision, 2009 IEEE 12th International Conference on. pp. 1451–1458. IEEE (2009)